Advance Smart Surveillance System

K. Aravindan*, M. Kanniyappan** & R. Karthi***

*Guided by G. Kumuran, Assistant Professor, Department of Electronics and Communication Engineering, SKP Engineering College, INDIA.
**Department of Electronics and Communication Engineering, SKP Engineering College, Tamil Nadu, INDIA.
***Department of Electronics and Communication Engineering, SKP Engineering College, Tamil Nadu, INDIA.

***Abstract*—**Surveillance and monitoring has become very important for security reasons these days. Residential areas, government organizations, commercial spaces, schools and hospitals, industries, banking and other challenging indoor and outdoor environments require high end surveillance systems, which are very expensive. This paper proposes the motion detection and tracking system for surveillance in this paper. The proposed system uses Raspberry Pi and PIR sensor and computer vision using SimpleCV to detect moving objects in the surveillance area, send alert to the user and streams the camera feed online using MPJG Streamer, which can be viewed by any authorized person on the go.

***Keywords*—**Camera; Computer Vision; PIR Sensor; Raspberry Pi; SimpleCV.

***Abbreviations*—**General-Purpose Input/Output (GPIO); High-Definition Multimedia Interface (HDMI); Universal Serial Bus (USB).

I. INTRODUCTION

TRADITIONAL surveillance system [Laur, 3] is enabled by high end cameras, video servers, network switch and monitoring computers. All these resources lead to complexity, high expense, and high power consumption and also require more area to establish. Also the feed can be viewed only at one particular area and cannot be accessed if the person is in motion. In the new surveillance system proposed, Raspberry Pi [Charles Severence, 2; Eben Upton & Gareth Halfacree, 7] operates and controls motion detectors and video cameras for remote sensing and surveillance, streams live video and records it for future playback. Computer Vision technique used is SimpleCV, which detects motion in the surveillance area and streams the live feed online using MPJG Streamer and alerts the authorized personnel via a message or email or an alarm. The authorized person can check the live feed from wherever he is located by just logging in online. According to the characteristic of the system, such as small size, low power consumption, quick speed and so on, it proves to be a very efficient surveillance system. The rest of the paper is structured as follows. Section II explains the functional description of all the components. Section III explains the system design. Section IV deals with algorithm of the program. Section V shows the results. Section VI discusses the conclusion and future work.

II. FUNCTIONAL DESCRIPTION

The functions of the various components are given below:

2.1. USB Camera

USB Cameras are imaging cameras that use USB 2.0 or USB 3.0 technology to transfer image data. USB cameras are designed to easily interface with dedicated computer systems by using the same USB technology that is found on most computers. The camera model used here is USB Camera model 2.0. The accessibility of USB technology in computer systems as well as the 480 Mb/s transfer rate of USB 2.0 makes USB Cameras ideal for many imaging applications.

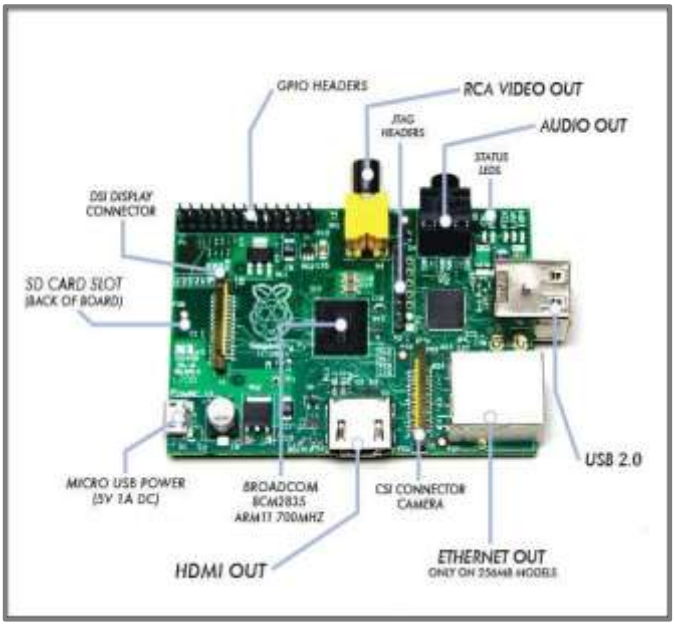

Figure 1: USB Camera Model 2.0

2.2. Raspberry Pi

The Raspberry Pi [Gareth Mitchell, 1; Matt Richardson & Shawn Wallace, 5] is a low cost single board, packing considerable computer power in a size of a credit card. The Raspberry Pi board contains many features like camera connector, Ethernet port, GPIO pins for interfacing sensors and switches, USB ports to connect to external devices(like keyboard, mouse, Wi-Fi adapter etc.,), HDMI port to interface to monitors (like LCD screens, projectors, TVs etc.) and an audio jack also available. By all these embedded on a single board. The Raspberry Pi has no internal mass storage or built-in operating system and hence it requires an SD card preloaded with a version of the Linux Operating System. Refer the figure 2.

This system proposed here uses Raspberry Pi [9] Model B+, shown in Figure 2. This model board is a micro-controller kit with in-built ARM11 processor provided with internet/Ethernet connectivity, dual USB connector, 512MB memory and supports Linux operating systems like Raspbian, Pidora, Raspbmc etc.

2.3. SimpleCV for Motion Detection

Computer vision is moving from a niche tool to an increasingly common tool for a diverse range of applications, such as facial recognition programs, automotive safety systems, industrial automation, biometrics, planetary explorations etc. One of the things that makes it feasible is that these days, the hardware requirements are inexpensive enough to allow more casual developers entry into the field, opening the door to many new applications and innovations.

We use SimpleCV [10] technique here in this surveillance system for motion detection in the area of surveillance. SimpleCV is an open source framework for building computer vision applications. It is a collection of libraries and software that can be used to develop vision applications. It provides the feasibility to work with the images or video streams that come from USB Cameras, webcams, Kinects, FireWire and IP cameras or mobile phones. SimpleCV is written in Python, and it's free to use. It runs on Mac, Windows, and Ubuntu Linux, and it is licensed under the BSD license.

2.4. M-JPG Streamer for Video Streaming

There are few modern streaming protocols for web browsers like HLS for Apple products, Fragmented MP4 etc.. But HLS supports only iDevices, but not much elsewhere. Fragmented MP4 is supported by Adobe and Microsoft, but requires browser plugins from these companies on the player computer, so Windows and Mac computers can do it, but Linux and mobile cannot.

MJPG-streamer is a command line application that copies JPG-frame from a single input plug-in to multiple outputs plug-in. It can be used to stream JPEG files over an IP-based network from the webcam to a viewer like Firefox, Video LAN client or even to a Windows Mobile device running the TCPMP-Player. It was written for embedded devices with very limited resources in terms of RAM and CPU.

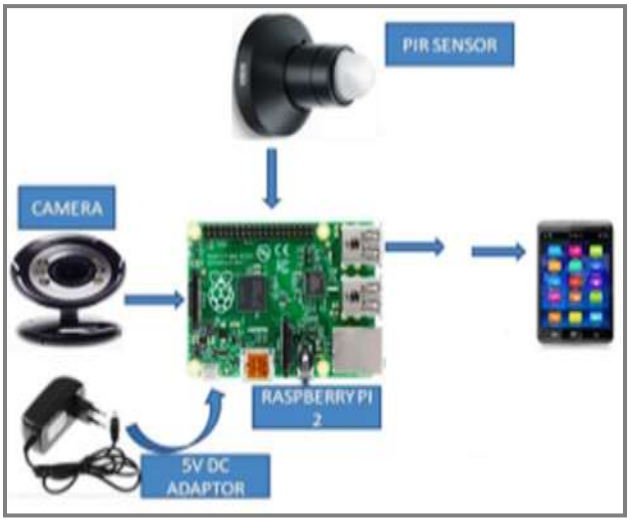

Figure 2: Proposed System

III. IMPLEMENTATION

3.1. System Design

The basic aim of system design is to continuously capture the surroundings under surveillance and if there is any moment is noticed, it turns on the lights and captures the screenshots and sends those over internet and it also alerts the authorized persons about human presence.

A program is written for continuously capturing the surroundings using camera and comparing them with the image frame of time shift. This comparison detects the motion and to switch on lights. After switching on lights it takes screen shots and they are transferred over internet. A storage device is connected to raspberry pi through USB port to store the screenshots.

This system uses MPJG-Streamer to stream the video on monitor connected to Raspberry Pi through HDMI port. And this is also used to stream the video online i.e., we can see the live streaming anywhere over internet. We can connect it to several monitors at a time using HDMI extension switch.

IV. ALGORITHM

In Raspberry Pi [Chris Edwards, 4; Peter Membrey & David Hows, 6] Raspian operating system is installed. This Operating System is a Linux based; it supports all programming languages like Python, C etc. Python programming language is used in the system to communicate with General Purpose Input Output ports and easy connection with databases using MySQLdb and SimpleCV modules.

In Raspberry Pi a program is written in python [8] using SimpleCV module to capture the successive images, finding the motion and alerting the user.

By following the below steps the system is implemented:

- First import all the modules required to for GPIO and SimpleCV

- Communicate with the camera connected to Raspberry Pi with SimpleCV
- Capture the successive images/videos with SimpleCV using camera
- Compare the images/videos to find the human presence
- If motion is detected, switch on the lights and take screenshots of surrounding area
- Transfer the screenshots over internet and store them in local storage
- Alerting the user about human presence
- If motion is not detected, then surveillance continued

Figure 3: Working

V. RESULTS

Surveillance system is practically implemented and the results are obtained. Results of ARS system are as follows: A sequence of images showing the results of our algorithm by comparing successive images\videos captured by camera to detect motion. After detection of motion shows the send alert to the user and transferring the screenshots over internet and shows the website with live feed from cameras connected to Raspberry Pi.

VI. CONCLUSION

In this paper, we have designed and implemented a low-cost and efficient Surveillance System capable of recording/capturing video/image and transmitting to the internet. It is advantageous as it offers reliability and privacy on both sides. It is authenticated and encrypted on the receiver side; hence it offers the sensitive information only to the authorized personnel. Also because of its small size and portability, it can be placed in any kind of surrounding for surveillance. Areas where movement is restricted such as sensitive military/nuclear sites or in banks, this kind of surveillance and monitoring systems can be implemented. This will not only omit the requirement of physical presence for surveillance, but will also capture all information in the storage which will be proving very beneficial.

REFERENCES

[1] Gareth Mitchell (2012), "The Raspberry Pi Single-Board Computer will Revolutionize Computer Science Teaching [For & against]", Vol. 7, No. 3, Pp. 26.
[2] Charles Severence (2013), "Eben Upton: Raspberry Pi", Vol. 46, No. 10, Pp. 1416.
[3] I. Laur (2010), "Microcontroller based Automation System with Security", *International Journal of Advanced Computer Science and Applications*, Vol. 1, No. 6, Pp. 60–65.
[4] Chris Edwards (2013), "Not-so-Humble Raspberry pi gets Big Ideas", Vol. 8, No. 3, Pp. 30–33.
[5] Matt Richardson & Shawn Wallace (2013), "Getting Started with Raspberry Pi", *United States of America: O'Reilly Media*.
[6] Peter Membrey & David Hows (2012), "Learn Raspberry Pi with Linux", *New York City: Apress*, Pp. 1–149.
[7] Eben Upton & Gareth Halfacree (2012), "Raspberry Pi User Guide", *A John Wiley and Sons Ltd*.
[8] Python Software Foundation [US], https://pypi.python.org/pypi
[9] Raspberry Pi Foundation, http://www.raspberry.org
[10] SimpleCV. http://www.simplecv.

